Ty Limpasuvan

ACM 104 Homework 3

① Ⓐ It is possible to reconstruct an inner product from the norm.

$$\|u+v\|^2 = \langle u+v, u+v\rangle = \langle u, u+v\rangle + \langle v, u+v\rangle$$

$$= \langle u,u\rangle + \langle u,v\rangle + \langle v,u\rangle + \langle v,v\rangle$$

$$= \langle u,u\rangle + 2\cdot\langle u,v\rangle + \langle v,v\rangle = \|u^2\| + 2\langle u,v\rangle + \|v\|^2$$

Which gives:

$$\langle u,v\rangle = \frac{\|u+v\|^2 - \|u\|^2 - \|v\|^2}{2}$$

Ⓑ We can show by contradiction that there's only one inner product that can generate a norm. We assume that there are two distinct inner products $\langle\cdot,\cdot\rangle_1$ and $\langle\cdot,\cdot\rangle_2$ that generate the same norm but aren't identical.

We then select vectors u and v such that $\langle u,v\rangle_1 \neq \langle u,v\rangle_2$. From the definition of $\langle\cdot,\cdot\rangle_1$ and $\langle\cdot,\cdot\rangle_2$ plus the norm,

$$\sqrt{\langle v,v\rangle_1} = \sqrt{\langle v,v\rangle_2} \text{ for } v \in V$$

and

$$\langle u+v, u+v\rangle_1 = \langle u+v, u+v\rangle_2 \quad \text{for } u,v \in V$$

Note that $\langle u+v, u+v\rangle = \langle u,u\rangle + 2\cdot\langle u,v\rangle + \langle v,v\rangle$ (See part Ⓐ)

Using this, we get:

$$\langle u+v, u+v\rangle_1 = \langle u,u\rangle_1 + 2\cdot\langle u,v\rangle_1 + \langle v,v\rangle_1$$

$$\langle u+v, u+v\rangle_2 = \langle u,u\rangle_2 + 2\cdot\langle u,v\rangle_2 + \langle v,v\rangle_2$$

$$\langle u+v, u+v\rangle_1 - \langle u+v,u+v\rangle_2 = \langle u,u\rangle_1 + 2\cdot\langle u,v\rangle_1 - 2\cdot\langle u,v\rangle_2 + \langle v,v\rangle_1 - \langle v,v\rangle_2$$

$$2\langle u,v\rangle_2 = 2\cdot\langle u,v\rangle_1 \implies \langle u,v\rangle_2 = \langle u,v\rangle_1$$

This conclusion contradicts our assumption; it implies both expressions are the same inner product. Thus, the inner product generating a norm must be unique.

② Ⓐ $\langle f,g\rangle_2$ defines an inner product on $C'[0,1]$.

$\langle f,g\rangle_1$ is not an inner product; it's not positive definite.

If we have $f(x) = C$ (some constant C), $f'(x)=0$.

This means $\langle f,f\rangle_1 = \int_0^1 f(x) f'(x)dx = \int_0^1 0\, dx = 0$

however, f is not the zero vector. $\langle f,g\rangle_2$, on the other hand, is bilinear, symmetric, and positive-definite.

Ⓑ $$\langle f,g\rangle = \int_0^1 (f(x)g(x) + f'(x)g'(x))dx$$

Cauchy-Schwarz:

$$\sqrt{\int_0^1 (f(x)g(x) + f'(x)g'(x))\, dx} \leq \sqrt{\int_0^1 (f(x)^2 + f'(x)^2)dx} \cdot \sqrt{\int_0^1 (g(x)^2 + g'(x)^2)dx}$$

Triangle:

$$\sqrt{\int_0^1 ((f(x)+g(x))^2 + (f'(x)+g'(x))^2)dx} \leq \sqrt{\int_0^1 (f(x)^2 + f'(x)^2)dx} + \sqrt{\int_0^1 (g(x)^2 + g'(x)^2)dx}$$

Ⓒ $$\cos\theta = \frac{\int_0^1 (f(x)g(x) + F'(x)g'(x))dx}{\sqrt{\int_0^1 (f(x)^2 + f'(x)^2)dx} \cdot \sqrt{\int_0^1 (g(x)^2 + g'(x)^2)dx}}$$

$$= \frac{\int_0^1 (e^x + 0)dx}{\sqrt{\int_0^1 (1^2 + 0^2)\,dx} \cdot \sqrt{\int_0^1 (e^{2x} + e^{2x})dx}} = \frac{e-1}{\sqrt{1} \cdot \sqrt{e^2-1}} = \frac{\sqrt{e-1}}{\sqrt{e+1}}$$

$\theta = 0.823$ radians

Ty Limpasuvan

ACM 104 Homework 3 Part 2

③ Attached.
Minimum value achieved in last iteration for ρ: 1014.6506

④ Ⓐ $L^2 = \int_0^1 f(x)g(x)\,dx$

$$G = \begin{bmatrix} \langle 1,1\rangle & \langle 1,e^x\rangle & \langle 1,e^{2x}\rangle \\ \langle e^x,1\rangle & \langle e^x,e^x\rangle & \langle e^x,e^{2x}\rangle \\ \langle e^{2x},1\rangle & \langle e^{2x},e^x\rangle & \langle e^{2x},e^{2x}\rangle \end{bmatrix} = \begin{bmatrix} \int_0^1 1dx & \int_0^1 e^x dx & \int_0^1 e^{2x} dx \\ \int_0^1 e^x dx & \int_0^1 e^{2x} dx & \int_0^1 e^{3x} dx \\ \int_0^1 e^{2x} dx & \int_0^1 e^{3x} dx & \int_0^1 e^{4x} dx \end{bmatrix}$$

$$= \begin{bmatrix} 1 & e-1 & \frac{1}{2}(e^2-1) \\ e-1 & \frac{1}{2}(e^2-1) & \frac{1}{3}(e^3-1) \\ \frac{1}{2}(e^2-1) & \frac{1}{3}(e^3-1) & \frac{1}{4}(e^4-1) \end{bmatrix}$$

Ⓑ Yes, G is positive definite. The terms 1, e^x, and e^{2x} are linearly independent.

Ⓒ $$G_2 = \begin{bmatrix} \langle 1,1\rangle_2 & \langle 1,e^x\rangle_2 & \langle 1,e^{2x}\rangle_2 \\ \langle e^x,1\rangle_2 & \langle e^x,e^x\rangle_2 & \langle e^x,e^{2x}\rangle_2 \\ \langle e^{2x},1\rangle_2 & \langle e^{2x},e^x\rangle_2 & \langle e^{2x},e^{2x}\rangle_2 \end{bmatrix} = \begin{bmatrix} \int_0^1 1dx & \int_0^1 2\cdot e^x dx & \int_0^1 3\cdot e^{2x} dx \\ \int_0^1 2\cdot e^x dx & \int_0^1 2\cdot e^{2x} dx & \int_0^1 3\cdot e^{3x} dx \\ \int_0^1 3\cdot e^{2x} dx & \int_0^1 3\cdot e^{3x} dx & \int_0^1 5\cdot e^{4x} dx \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 2e-2 & \frac{3}{2}(e^2-1) \\ 2e-2 & e^2-1 & e^3-1 \\ \frac{3}{2}(e^2-1) & e^3-1 & \frac{5}{4}(e^4-1) \end{bmatrix}$$

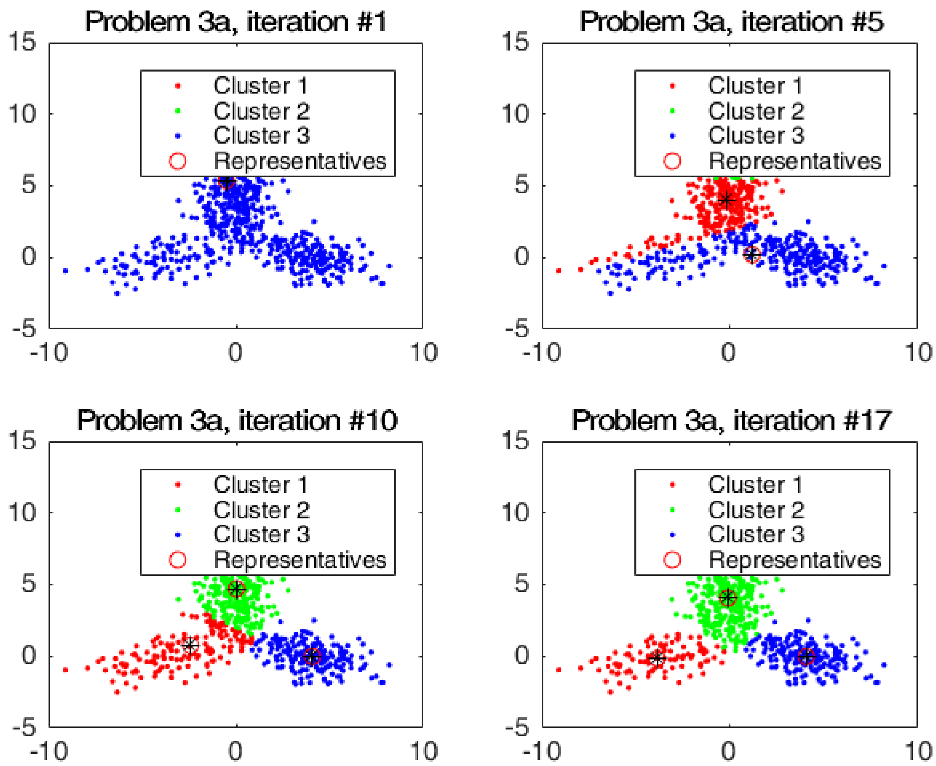
Ⓓ Because 1, e^x, and e^{2x} are linearly independent, the Gram matrix generated using any inner product for that vector space would be positive-definite. Thus, $\langle\cdot,\cdot\rangle$ and $\langle\cdot,\cdot\rangle_2$ cannot be found to have exactly one produce a positive-definite Gram matrix.

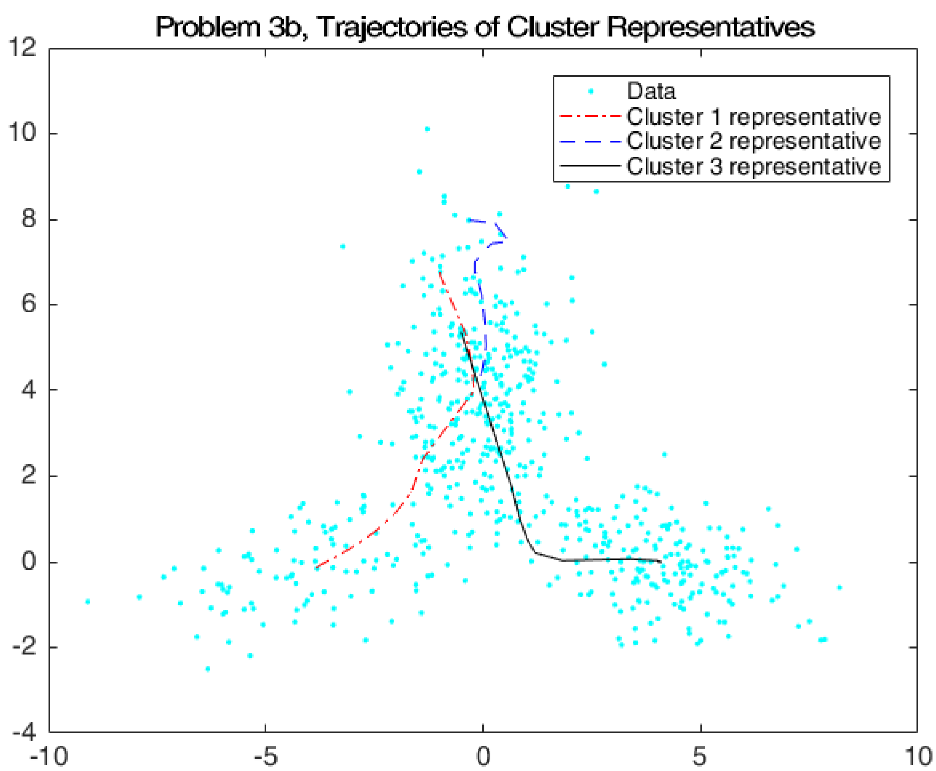
⑤ Attached. According to my results, Ted Cruz's page is most similar to the constitution's text.

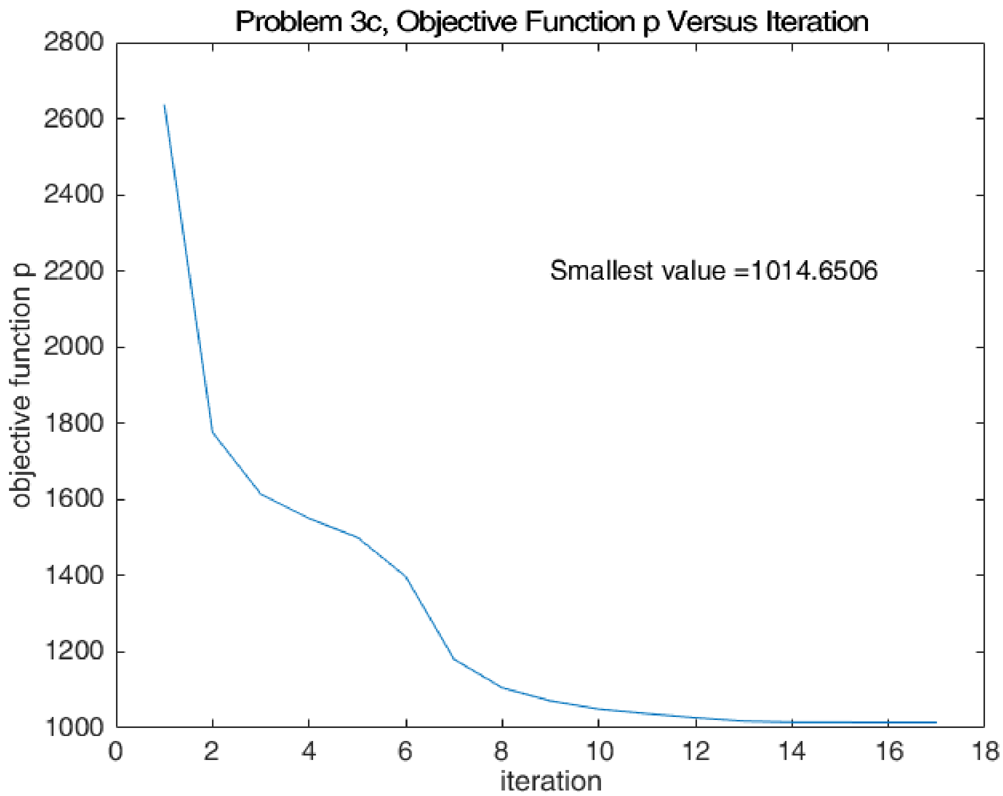
```
%Problem 3

load('clustering_data.mat');
rng(2016);
[m, n] = size(x);

K = 3;

clust_assignment = zeros(m, 1);
clust1 = zeros(0, 2);
clust2 = zeros(0, 2);
clust3 = zeros(0, 2);

trajectory1 = zeros(0, 2);
trajectory2 = zeros(0, 2);
trajectory3 = zeros(0, 2);

objective_per_iteration = zeros(0, 2);

%Part A
figure
subplot_id = 1;
rs = datasample(x, K);
iteration = 1;
while 1
    clust1 = zeros(0, 2);
    clust2 = zeros(0, 2);
    clust3 = zeros(0, 2);
    change_occured = 0;
    p_sum = 0;
    for i = 1:m
        vector = x(i, :);
        norm1 = norm(vector - rs(1, :));
        norm2 = norm(vector - rs(2, :));
        norm3 = norm(vector - rs(3, :));
        min_norm = min([norm1, norm2, norm3]);
        p_sum = p_sum + min_norm;
        new_clust = 0;
        if min_norm == norm1
            new_clust = 1;
            clust1 = [clust1; vector];
        end
        if min_norm == norm2
            new_clust = 2;
            clust2 = [clust2; vector];
        end
        if min_norm == norm3
            new_clust = 3;
            clust3 = [clust3; vector];
        end
        if new_clust ~= clust_assignment(i)
            change_occured = 1;
```

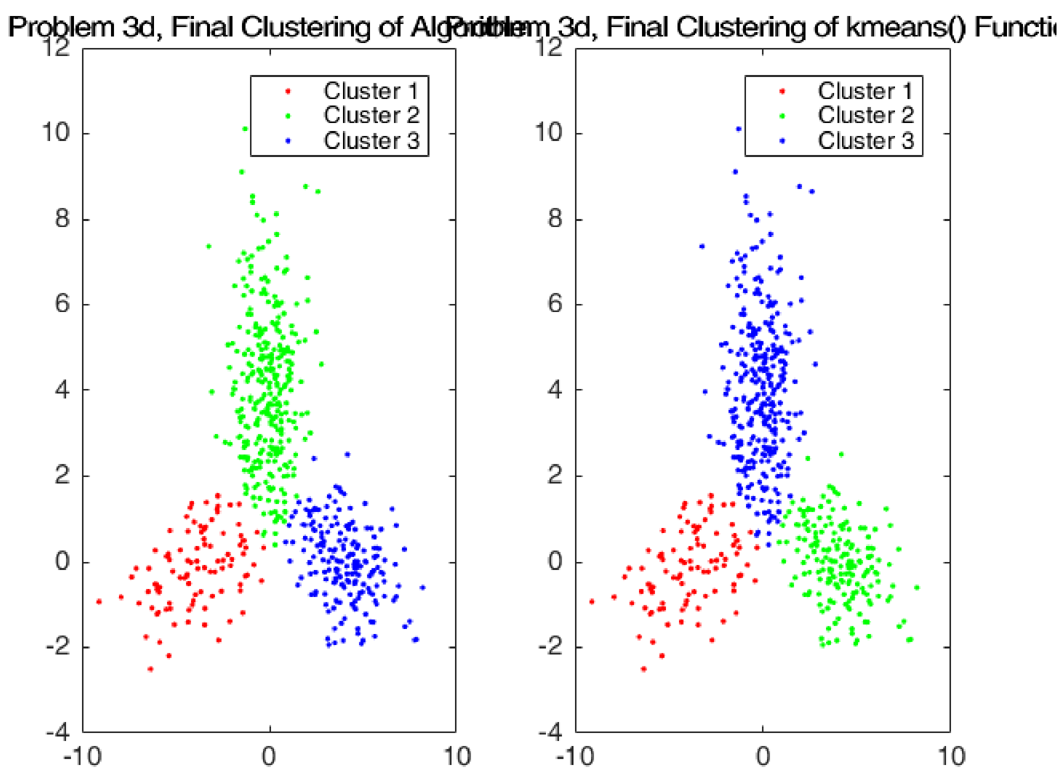
```
        end
        clust_assignment(i) = new_clust;
    end
    p_row = [iteration p_sum];
    objective_per_iteration = [objective_per_iteration; p_row];

    %Plots for 1st, 5th, 10th, and last iterations
    if iteration == 1 || iteration == 5 || iteration == 10 ||
 change_occured == 0
        sp = subplot(2, 2, subplot_id);
        subplot_id = subplot_id + 1;
        plot(sp, clust1(:, 1), clust1(:, 2), '.r', clust2(:, 1),
 clust2(:, 2), '.g', clust3(:, 1), clust3(:, 2), '.b', rs(:, 1), rs(:,
 2), 'or', rs(:, 1), rs(:, 2), '*k');
        legend('Cluster 1', 'Cluster 2', 'Cluster
 3', 'Representatives')
        title(strcat('Problem 3a, iteration #', int2str(iteration)));
    end
    iteration = iteration + 1;

    mean1 = mean(clust1);
    mean2 = mean(clust2);
    mean3 = mean(clust3);
    if isequal(rs(1), mean1) || isequal(rs(2), mean2) ||
 isequal(rs(3), mean3)
        change_occured = 1;
    end
    trajectory1 = [trajectory1; rs(1, :)];
    trajectory2 = [trajectory2; rs(2, :)];
    trajectory3 = [trajectory3; rs(3, :)];
    rs(1, :) = mean1;
    rs(2, :) = mean2;
    rs(3, :) = mean3;

    %Break for convergence
    if change_occured == 0
        break;
    end
end

%Part B
figure
plot(x(:, 1), x(:, 2), '.c', trajectory1(:, 1), trajectory1(:,
 2), '-.r', trajectory2(:, 1), trajectory2(:, 2), '--b',
 trajectory3(:, 1), trajectory3(:, 2), '-k')
legend('Data', 'Cluster 1 representative', 'Cluster 2
 representative', 'Cluster 3 representative');
title('Problem 3b, Trajectories of Cluster Representatives')

%Part C
figure
plot(objective_per_iteration(:, 1), objective_per_iteration(:, 2))
title('Problem 3c, Objective Function p Versus Iteration')
xlabel('iteration')
```

```
ylabel('objective function p')
last_row = objective_per_iteration(end, :);
text(9, 2200, strcat('Smallest value = ', num2str(last_row(2))));

%Part D
figure
sp = subplot(1, 2, 1);
plot(sp, clust1(:, 1), clust1(:, 2), '.r', clust2(:, 1), clust2(:,
 2), '.g', clust3(:, 1), clust3(:, 2), '.b');
legend('Cluster 1', 'Cluster 2', 'Cluster 3')
title('Problem 3d, Final Clustering of Algorithm');
sp = subplot(1, 2, 2);
idx = kmeans(x, K);
clust1x = zeros(0, 2);
clust2x = zeros(0, 2);
clust3x = zeros(0, 2);
for i = 1:m
    vector = x(i, :);
    if idx(i) == 1
        clust1x = [clust1x; vector];
    end
    if idx(i) == 2
        clust2x = [clust2x; vector];
    end
    if idx(i) == 3
        clust3x = [clust3x; vector];
    end
end
plot(sp, clust1x(:, 1), clust1x(:, 2), '.r', clust2x(:, 1), clust2x(:,
 2), '.g', clust3x(:, 1), clust3x(:, 2), '.b');
legend('Cluster 1', 'Cluster 2', 'Cluster 3')
title('Problem 3d, Final Clustering of kmeans() Function');
```


Problem 3a, iteration #1
Problem 3a, iteration #5
Problem 3a, iteration #10
Problem 3a, iteration #17
Cluster 1
Cluster 2
Cluster 3
Representatives


Problem 3b, Trajectories of Cluster Representatives
Data
Cluster 1 representative
Cluster 2 representative
Cluster 3 representative


Problem 3c, Objective Function p Versus Iteration
objective function p
iteration
Smallest value =1014.6506
2800
2600
2400
2200
2000
1800
1600
1400
1200
1000
0
2
4
6
8
10
12
14
16
18


Problem 3d, Final Clustering of Algorithm
Problem 3d, Final Clustering of kmeans() Functi
Cluster 1
Cluster 2
Cluster 3
12
10
8
6
4
2
0
-2
-4
-10
0
10

Published with MATLAB® R2016a

```
% Problem 5
%
% Using these 10 key words for dictionary:
% w1  = president
% w2  = laws
% w3  = united states
% w4  = citizens
% w5  = power
% w6  = constitution
% w7  = duties
% w8  = congress
% w9  = senators
% w10 = executive

constitution = [ 120 13 86 18 36 27 16 60 13 13 ];
constitution = constitution/norm(constitution);
bernie = [ 62 3 45 6 5 1 0 39 10 0 ];
bernie = bernie/norm(bernie);
hilary = [ 146 2 51 10 31 4 2 27 10 0 ];
hilary = hilary/norm(hilary);
trump = [ 216 14 61 9 10 1 0 9 0 30 ];
trump = trump/norm(trump);
tedcruz = [ 82 7 46 18 9 11 0 26 15 2 ];
tedcruz = tedcruz/norm(tedcruz);
kasich = [ 58 9 18 7 9 5 0 26 2 9 ];
kasich = kasich/norm(kasich);

Z = zeros(0, 10);
Z = [Z; constitution];
Z = [Z; bernie];
Z = [Z; hilary];
Z = [Z; trump];
Z = [Z; tedcruz];
Z = [Z; kasich];
Z = Z';
G = Z' * Z;
index = 0;
max = 0;
for i = 2:6
    cos = G(i, 1);
    if cos > max
        max = cos;
        index = i;
    end
end

% print results
if index == 2
    fprintf('Bernie\n');
end
if index == 3
    fprintf('Hillary\n');
```

```
end
if index == 4
    fprintf('Trump\n');
end
if index == 5
    fprintf('Ted Cruz\n');
end
if index == 6
    fprintf('Kasich\n');
end
```

```
Ted Cruz
```

Published with MATLAB® R2016a